REMARKS

Claims 1 and 3-13 were pending in the application. New claims 14-16 have been added. Applicants have amended claims 1, 4-8, and 10 and cancelled claims 12 and 13 from further consideration in this application. Applicants are not conceding in this application that those claims are not patentable over the art cited by the Examiner. Applicants respectfully reserve the right to pursue these and other claims in one or more continuations and/or divisional patent applications. Applicant respectfully requests reconsideration.

CLAIM REJECTIONS UNDER 35 USC §112

The Office Action rejected claims 1 and 3-13 under 35 USC 112, first paragraph, for failing to comply with the written description requirement.

Claim 1 has been amended. Applicant submits that these changes moot the 112 rejections. Support for the claim amendments can be found in paragraphs [0027], [0090]-[0095], and [0109] – [0117] of Applicant's specification. These paragraphs describe the method of claim 1; specifically, they describe how it is sufficient to periodically examine the references from the array, to a depth of one, rather than keeping "track of every element in those leaking data structures" -- the former is a constant-sized traversal, the latter (every element) is one that is unbounded in size.

CLAIM REJECTIONS UNDER 35 USC §102

The Office Action rejected claims 1 and 3-13 under 35 USC 102(b) as being anticipated by Bournas et al. (US 6,061,679, hereinafter "Bournas").

Claim 1 has been amended to more clearly track the key aspect of the invention, which is: for evolving data structures, we need only perform single-depth traversals of the evolving data structures in order to detect evolutionary trends in the region in which the data structures are located.

Bournas does not teach the elements of amended claim 1:

deriving a suspect region from a running application;
periodically traversing only selected subgraphs to a depth of one;
characterizing the data structure changes;
updating a histogram of the suspect region;
using the characterized data structure changes to describe, characterize, and identify an evolutionary trend of the suspect region as a whole; and
reporting the characterized changes to the region to an analysis agent.

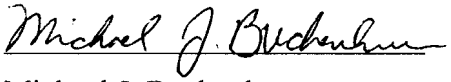
Bournas's method for creating and searching a data structure uses an ordering scheme comprising a tree structure. [Bournas, Col. 5, lines 1-3] "On the other hand, if the key mask is stored within the range, then a find procedure is used to locate the key in the found tree, STEP 518." [Bournas, col. 10, 11-13] "Described in detail above are examples of techniques for adding, deleting and searching a data structure…" [Bournas, col. 11, 19-

20]

Bournas does not perform a traversal limited to a depth of one. Bournas must traverse an entire graph, because Bournas describes what amounts to an efficient way to do lookups. In performing a lookup in a binary tree, one traverses the tree to find the matching element. The number of nodes visited is log(N), for a graph with N nodes. This is not constant-sized.

For the foregoing reasons, Applicant respectfully requests allowance of the pending claims.

Respectfully submitted,



Michael J. Buchenhorner
Reg. No. 33,162

Date: May 4, 2008

Michael Buchenhorner, P.A.
8540 S.W. 83 Street
Miami, Florida 33143
(305) 273-8007 (voice)
(305) 595-9579 (fax)